

B.4 typing

Reference: <https://docs.python.org/3.9/library/typing.html>.

Type	Description
Any	A value that could be of any type. (Used as a placeholder when a variable's type could be anything, or is unknown.)
Callable[[T1, T2, ...], Tr]	A function whose parameters have type T1, T2, etc., and whose return type is Tr. Example: the function <pre>def f(x: int, y: str) -> bool: # Body omitted</pre> has type Callable[[int, str], bool].
dict[T1, T2]	A dictionary whose keys have type T1 and whose values have type T2. Example: {'a': 1, 'b': 2} has type dict[str, int].
list[T]	A list whose elements all have type T. Example: [1, 2, 3] has type list[int].
Optional[T]	Synonym of Union[T, None].
set[T]	A set whose elements all have type T. Example: {'hi', 'bye'} has type set[str].

Type	Description
<code>tuple[T1, T2, ...]</code>	<p>A tuple whose first element has type <code>T1</code>, second element has type <code>T2</code>, etc.</p> <p>Example: <code>('hello', True, 3.4)</code> has type <code>tuple[str, bool, float]</code>.</p>
<code>Union[T1, T2, ...]</code>	<p>A value whose type is one of <code>T1</code>, <code>T2</code>, etc.</p> <p>Example: both <code>1</code> and <code>'hello'</code> are instance of type <code>Union[int, str]</code>.</p>